CLAIMS

1. A method of performing memory mapped input output operations to an alternate address space comprising:

establishing a first instruction directed to a first memory mapped input output alternate address space associated with an adapter to store data in accordance with a definition of a z/Architecture;

establishing a second instruction directed to said first memory mapped input output alternate address space associated with an adapter to load data in accordance with said definition(s) of said z/Architecture; and

wherein a process issues at least one of said first instruction and said second instruction and thereby causes execution of at least one of said store and load with said first alternate address space.

2. The method of Claim 1 wherein said first alternate address space is not a partition of a main address space from which said issuing process is executing.

3. The method of Claim 1 wherein said process issuing said at least one of said first instruction and said second instruction and thereby causes execution of at least one of said store and load with said first alternate address space operates in a problem state of a machine

4. The method of Claim 1 wherein said execution includes said at least one of said store and load with an allocated resources associated first alternate address space

5. The method of Claim 1 wherein said problem state corresponds to a least privileged execution state in said z/Architecture

6. The method of Claim 1 wherein said first alternate address space is associated with an adapter and alleviates use of a main address space of said process or of another adapter.

7. The method of Claim 1 wherein at least one of said first instruction and said second instruction is executed without supervisory state intervention.

8. The method of Claim 1 wherein said first instruction and said second instruction are semiprivileged instructions that may be executed in problem state, wherein ownership of a specified resource of a specified adapter determines a privilege required for execution of said semiprivileged instructions.

9. The method of Claim 1 further including a second alternate address space associated with a second adapter.

10. The method of Claim 9 wherein a storage location in said first alternate address space maps to a different address than the same location in said second alternate address space.

11. The method of Claim 1 wherein said adapter includes address spaces as partitions of said alternate address space.

12. The method of Claim 11 wherein said multiple address spaces are governed by at least one of a resource type and storage area types associated with said adapter.

13. A method of performing memory mapped input output operations to an alternate address space comprising:

establishing a first instruction directed to a first memory mapped input output alternate address space associated with an adapter to store;

establishing a second instruction directed to said first memory mapped input output alternate address space associated with an adapter to load data; and

wherein a process issues at least one of said first instruction and said second instruction and thereby causes execution of at least one of said store and load with said first alternate address space and operates in a problem state of a machine.

14. The method of Claim 13 wherein said problem state corresponds to a least privileged execution state in said z/Architecture

15. The method of Claim 13 wherein said first instruction and said second instruction are semi-privileged instructions, wherein ownership of a specified resource of a specified adapter determines a privilege required for execution of said semi-privileged instructions.

16. The method of Claim 13 further including a second alternate address space associated with a second adapter.

17. The method of Claim 16 wherein a storage location in said first alternate address space maps to a different address than the same location in said second alternate address space.

18. A storage medium encoded with a machine-readable computer program code, said code including instructions for causing a computer to implement a method of performing memory mapped input output operations to an alternate address space, the method comprising:

establishing a first instruction directed to a first memory mapped input output alternate address space associated with an adapter to store data in accordance with a definition of a z/Architecture;

establishing a second instruction directed to said first memory mapped input output alternate address space associated with an adapter to load data in accordance with said definition(s) of said z/Architecture; and

wherein a process issues at least one of said first instruction and said second instruction and thereby causes execution of at least one of said store and load with said first alternate address space.

19. A system for performing memory mapped input output operations to an alternate address space comprising:

a means for establishing a first instruction directed to a first memory mapped input output alternate address space associated with an adapter to store data in accordance with a definition of a z/Architecture;

a means for establishing a second instruction directed to said first memory mapped input output alternate address space associated with an adapter to load data in accordance with said definition(s) of said z/Architecture; and

wherein a process issues at least one of said first instruction and said second instruction and thereby causes execution of at least one of said store and load with said first alternate address space.